



Journal of Statistical Software

June 2009, Volume 30, Issue 11.

<http://www.jstatsoft.org/>

Multi-Objective Optimal Experimental Designs for ER-fMRI Using MATLAB

Ming-Hung Kao
University of Georgia

Abstract

Designs for event-related functional magnetic resonance imaging (ER-fMRI) that help to efficiently achieve the statistical goals while taking into account the psychological constraints and customized requirements are in great demand. This is not only because of the popularity of ER-fMRI but also because of the high cost of ER-fMRI experiments; being able to collect highly informative data is crucial. In this paper, we develop a MATLAB program which can accommodate many user-specified experimental conditions to efficiently find ER-fMRI optimal designs.

Keywords: design efficiency, genetic algorithms, MATLAB.

1. Introduction

Event-related functional magnetic resonance imaging (ER-fMRI) is one of the leading technologies for studying human brain activity in response to mental tasks or stimuli. Before conducting an ER-fMRI experiment, a sequence of stimuli of one or more types interlaced with the control (rest or fixation) is prepared. This sequence of stimuli is presented to an experimental subject while the MR scanner scans his/her brain every few seconds. The blood oxygenation level dependent (BOLD) time series is collected from each brain voxel (a small region of the brain) for statistical analysis. See [Josephs *et al.* \(1997\)](#), [Rosen *et al.* \(1998\)](#), [Dale \(1999\)](#), and [Bandettini and Cox \(2000\)](#) for overviews of ER-fMRI.

One important design problem of ER-fMRI is to find an optimal sequence of the stimuli best suited to the researcher's needs. However, this problem is difficult due to the following reasons. First, the design space, consisting of all possible sequences of stimuli, is enormous. Searching over this space for a good design is hard. In addition, the flexibility of ER-fMRI allows researchers to consider two popular statistical objectives, namely estimation and detection. Estimation refers to the estimation of the hemodynamic response function (HRF), a function

of time describing an effect of a single, brief stimulus. Detection is to identify brain regions that are activated by the stimuli. Considering both objectives in one study is not uncommon (see also [Wager and Nichols 2003](#)), but this requires good multi-objective designs that efficiently achieve these two competing goals. Moreover, statistics is not the only concern for ER-fMRI. Psychology plays an important, even crucial, role. When a design sequence is patterned or easy to predict, psychological effects such as habituation or anticipation can occur to contaminate the data ([Dale 1999](#)). A good design should help to avoid these confounds. Furthermore, customized requirements such as the required number of stimuli for each stimulus type can also arise to make the problem even more complicated. As a result, searching for a good multi-objective design is inevitable. We need well-defined multi-objective design criteria (MO-criteria) for evaluating the quality of competing designs, an efficient search algorithm and a program that implements such an algorithm.

In this paper, we develop a program using MATLAB ([The MathWorks, Inc. 2006](#)) for finding multi-objective optimal ER-fMRI designs. Our program implements the approach proposed by [Kao et al. \(2009a\)](#), which includes rigorously formulated models, well-defined MO-criteria and a genetic algorithm (GAs). They incorporate knowledge about the performance of well-known ER-fMRI designs to increase the effectiveness and efficiency of their approach. As demonstrated in their paper, this approach is more efficient than the previous methods and is flexible enough to accommodate different experimental conditions and assumptions. To make the best use of this approach, our program allows users to specify the experimental conditions based on their needs. The designs that we obtain can help researchers to achieve efficient statistical inference.

The rest of the article is organized as follows. Section 2 reviews the approach proposed by [Kao et al. \(2009a\)](#). Section 3 illustrates our computer codes. An example for using our program is in Section 4, followed by conclusion and discussion in Section 5.

2. Methodology review

A typical ER-fMRI design can be viewed as an alignment of events, including the stimuli and the control. For convenience, the symbols $0, 1, \dots, Q$ are used to represent the events with 0 indicating the control and i a type- i stimulus, $i = 1, \dots, Q$; Q is the total number of stimulus types. A design, denoted by ξ , looks like $\xi = \{101201210\dots1\}$.

While being presented to an experimental subject, each stimulus lasts for a short period of time relative to the inter-stimulus interval (ISI), the fixed time interval between the onsets of consecutive events. We note that 0s in the sequence are “pseudo-events”; they help to calculate the onset times of stimuli. For example, with a 0 in between, the first, second and the third stimuli (1, 1, and 2) of ξ occur, respectively, 1ISI, 3ISI, and 4ISI seconds after the outset of the experiment. The control fills up the time period between the end of a stimulus and the start of the next one.

Our goal is to find a best sequence of the events to efficiently achieve four popular objectives, which are 1) estimating the HRF, 2) detecting brain activation, 3) avoiding psychological confounds and 4) maintaining the desired frequency for each stimulus type. To define the design criteria for the first two statistical objectives, we need to specify the underlying models.

Models

Following previous approaches (e.g. [Liu 2004a](#); [Liu and Frank 2004](#); [Wager and Nichols 2003](#)), two popular linear models are considered for the two statistical objectives; see also [Friston et al. \(1995\)](#), [Worsley and Friston \(1995\)](#), and [Dale \(1999\)](#). These models are:

$$\mathbf{Y} = \mathbf{X}\mathbf{h} + \mathbf{S}\boldsymbol{\gamma} + \mathbf{e}; \quad (1)$$

$$\mathbf{Y} = \mathbf{Z}\boldsymbol{\theta} + \mathbf{S}\boldsymbol{\gamma} + \boldsymbol{\eta}, \quad (2)$$

where \mathbf{Y} is the voxel-wise BOLD time series, $\mathbf{h} = (\mathbf{h}'_1, \dots, \mathbf{h}'_Q)'$ is the parameter vector for the HRFs of the Q stimulus types, $\mathbf{X} = [\mathbf{X}_1 \cdots \mathbf{X}_Q]$ is the design matrix, $\boldsymbol{\theta} = (\theta_1, \dots, \theta_Q)'$ represents the response amplitudes, $\mathbf{Z} = \mathbf{X}\mathbf{h}_0$ is the convolution of stimuli with an assumed basis, \mathbf{h}_0 , of the HRF, $\mathbf{S}\boldsymbol{\gamma}$ is a nuisance term describing the trend or drift of \mathbf{Y} , and \mathbf{e} and $\boldsymbol{\eta}$ are noise. We assume a known whitening matrix, \mathbf{V} , such that $\mathbf{V}\mathbf{e}$ and $\mathbf{V}\boldsymbol{\eta}$ are white noise. The whitening matrix can be obtained empirically from previous experiments; see also [Wager and Nichols \(2003\)](#). Model (1) is typically used for estimating the HRF and model (2) is for detecting activation. Note that, for detection problems, a basis \mathbf{h}_0 for the HRF needs to be assumed.

To enable the use of a finite set of interpretable parameters to capture the fluctuation of the continuous HRF over time, the discretization interval ([Dale 1999](#)) is utilized for parameterizing the HRF in model (1). The length of the discretization interval, denoted by ΔT , is set to the greatest value dividing both the ISI and TR; the TR is the time interval between consecutive MR scans. The HRF parameters, captured in the vector \mathbf{h} , then represent the heights of the HRF for each stimulus after every ΔT seconds following the stimulus onset. This parametrization is explained in detail in [Kao et al. \(2009b\)](#).

Design criterion

For the two statistical objectives, two popular optimality design criteria, namely A - and D -optimality criteria ([Atkinson et al. 2007](#)), are considered. A -optimality aims at minimizing the average variance of estimators of parametric functions. A D -optimal design minimizes the generalized variance of estimators of linearly independent parametric functions, or, under normality, it minimizes the volume of simultaneous elliptical confidence regions for these parametric functions at any specified confidence level. The value of the design criterion for estimating the HRF, referred to as “estimation efficiency”, is denoted by F_e . Likewise, the term “detection power” and the notation F_d are used to indicate the value of the design criterion for detecting activation. These criteria have the forms of $F_i = r_c / \text{trace}(\mathbf{M})$ for A -optimality, or of $F_i = \det(\mathbf{M})^{-1/r_c}$ for D -optimality; $i = d, e$. Here, $\mathbf{M} = \mathbf{C}[\mathbf{W}'\mathbf{V}'(\mathbf{I} - P_{\mathbf{VS}})\mathbf{V}\mathbf{W}]^{-1}\mathbf{C}'$, $\mathbf{W} \equiv \mathbf{X}$ for F_e , $\mathbf{W} \equiv \mathbf{Z}$ for F_d , \mathbf{I} is an identity matrix, $P_{\mathbf{A}} = \mathbf{A}(\mathbf{A}'\mathbf{A})^{-1}\mathbf{A}'$ is the orthogonal projection on the vector space spanned by the column vectors of \mathbf{A} , \mathbf{A}^{-} is a generalized inverse matrix of \mathbf{A} , \mathbf{C} is a matrix of estimable linear combinations of the parameters, and r_c is the number of rows of \mathbf{C} . We note that F_e and F_d are “larger-the-better” criteria.

The third objective is to avoid psychological confounds. We would like a sequence that makes it difficult for a subject to anticipate future stimuli based on past stimuli. Designs minimizing the following criterion help to achieve this objective.

$$F_c = \sum_{r=1}^R \sum_{i=1}^Q \sum_{j=1}^Q [|n_{ij}^{(r)} - (n-r)P_i P_j|].$$

Here, the sub-design excluding all 0s but retaining all the stimuli of the original design is considered. In this design criterion, n is the length of the sub-design, and $n_{ij}^{(r)}$ is the number of times that i and j are r elements away in the sub-design; i.e., they are, respectively, the t th and the $(t + r)$ th elements, $t = 1, \dots, (n - r)$. P_i is the specified proportion for the type- i stimulus in the sub-design which may be taken as $1/Q$ if there is no preference, and $\lfloor |a| \rfloor$ is the integer part of the absolute value of a . R is a given integer; the unpredictability of a design increases when R increases. Therefore, F_c aims at having a design with each pair appearing a number of times that is proportional to the product of the specified proportions for the stimuli. Designs minimizing this criterion are said to be R th order counterbalanced (cf. [Wager and Nichols 2003](#)).

The fourth design criterion helps to maintain the desired stimulus frequency and is also defined on the sub-design. It is $F_f = \sum_{i=1}^Q \lfloor |n_i - nP_i| \rfloor$, where n_i is the number of the type- i stimulus in the sub-design. Designs achieving the desired stimulus frequency minimize F_f .

With these four individual design criteria, the family of MO-criteria is then defined as

$$\{F^* = w_c F_c^* + w_d F_d^* + w_e F_e^* + w_f F_f^* : w_i \geq 0, i = c, d, e, f; \sum_i w_i = 1\}, \quad (3)$$

where w_i s are weights selected based on the researcher's emphasis in a given study, $F_i^* = F_i / \max(F_i)$ for $i = d, e$ and $F_i^* = 1 - [F_i / \max(F_i)]$ for $i = c, f$. We note that, in [Kao et al. \(2009a\)](#), $\min(F_i)$ is also considered when calculating F_i^* . Since these minimal values are zero, they are omitted here. Each member of this family can serve as an objective function of the search algorithm.

Search algorithm

The search algorithm of [Kao et al. \(2009a\)](#) is built upon the genetic algorithm technique ([Holland 1975, 1992](#)). This technique is popular for solving optimization problems, in which good solutions (parents) are used to generate better ones (offsprings). To efficiently apply this technique to our problem, well-known results about good fMRI designs are incorporated so that the search over the huge design space can be carried out more efficiently. The outline of the algorithm is as follows:

- Step 1.** (Initial designs) Generate G initial designs consisting of random designs, an m -sequence-based design, a block design and their combinations. Use the objective function to evaluate the fitness of each initial design.
- Step 2.** (Crossover) With probability proportional to fitness, draw with replacement $G/2$ pairs of designs to crossover — select a random cut-point and exchange the corresponding fractions of the paired designs.
- Step 3.** (Mutation) Randomly select $q\%$ of the events from the G offspring designs. Replace these events by randomly generated ones. Here, an event is a stimulus or the control.
- Step 4.** (Immigration) Add to the population another I designs drawn from random designs, block designs and their combinations.
- Step 5.** (Fitness) Obtain the fitness scores of the offsprings and immigrants.

Step 6. (Natural selection) Keep the best G designs according to their fitness scores to form the parents of the next generation. Discard the others.

Step 7. (Stop) Repeat steps 2 through 6 until a stopping rule is met; e.g. after M_g generations. Keep track of the best design over generations.

In Step 1, m -sequences or m -sequence-based designs are generated following Liu (2004a); see also Buračas and Boynton (2002). These designs are well-known for their high estimation efficiencies. Good designs for detection are block designs. A block design is a sequence where stimuli of the same type are clustered into blocks. For example, a two-stimulus-type block design with a block size of four can consist of repetitions of {111122220000}. Repetitions of {1111000022220000} and {11112222} are other possible patterns. In steps 1 and 4, block designs with various block sizes and patterns are considered. A fraction of an m -sequence-based design or a random design is combined with a fraction of a block design to form a mixed design. These mixed designs along with random designs are also included as part of the initial designs and immigrants.

3. Code description

In this section, we describe our MATLAB program, including the input parameters, the way to run the program and the output variables.

3.1. Input parameters

Table 1 presents all the input parameters of our program along with their default values. Due to the complexity of ER-fMRI, a few parameters are needed for specifying experimental conditions and assumptions, which might vary from experiment to experiment. An example code, ‘Par_Assign.m’, is therefore provided to help to assign these parameter values. The values are specified through the MATLAB structure *Inp*. Some fields of *Inp* are detailed below. Descriptions for other input parameters can be found in Table 1.

Nuisance term

The nuisance term \mathbf{S} in (1) and (2) is specified in *Inp.Smat*. By default, \mathbf{S} corresponds to a second-order Legendre polynomial drift. The degree of the polynomial can be changed in ‘Par_Assign.m’ through the *PolyOrder* variable. While polynomial drift is popular (e.g. Worsley *et al.* 2002; Liu 2004a), other nuisance term can also be considered.

Whitening matrix

The square of the whitening matrix, \mathbf{V}^2 , described after (2) is specified in *Inp.V2*. By default, the following matrix, which corresponds to a stationary AR(1) process, is considered:

$$\mathbf{V}^2 = \begin{pmatrix} 1 & -\rho & 0 & \dots & 0 & 0 \\ -\rho & 1 + \rho^2 & -\rho & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 + \rho^2 & -\rho \\ 0 & 0 & 0 & \dots & -\rho & 1 \end{pmatrix}. \quad (4)$$

Parameter	Type	Description	Default
nSTYPE	integer	number of stimulus types (Q)	2
ISI	real	inter-stimulus interval; the time interval between event onsets (in seconds)	2.0
TR	real	time to repetition; the time interval between MR scans (in seconds)	2.0
dT	real	the discretization interval, $\Delta T =$ the greatest value dividing both the ISI and TR	2.0
nEvents	integer	number of events of the design	242
Smat	real matrix	the matrix \mathbf{S} in models (1) and (2)	2 nd -order polynomial
V2	real matrix	the square of the whitening matrix	see (4)
Opt	integer	0=A-optimality; 1=D-optimality	0
MOWeight	real array	weights of the four objectives, $[w_c, w_d, w_e, w_f]$	$[\frac{1}{4} \frac{1}{4} \frac{1}{4} \frac{1}{4}]$
basisHRF	real array	the basis function \mathbf{h}_0 for the HRF	canonical HRF of SPM2
durHRF	real	duration of the HRF (in seconds)	32.0
CX	real matrix	linear combinations of interest for the HRF parameters, $h_{ij}; i = 1, \dots, Q, j = 1, \dots, (1 + \lfloor \frac{K}{\Delta T} \rfloor)$	\mathbf{I}_{34}
CZ	real matrix	linear combinations of interest for the amplitudes, $\theta_i; i = 1, \dots, Q$	\mathbf{I}_2
MaxFe	real	$\max(F_e)$	39.2715
MaxFd	real	$\max(F_d)$	132.0670
cbalR	integer	the order of counterbalancing, R , for F_c	3
stimFREQ	real array	the desired frequency of each stimulus type P_i ; the array should sum to one	$[\frac{1}{2} \frac{1}{2}]$
sizeGen	integer	number of designs in each GA generation; an even number	20
qMutate	real	rate of mutation	0.01
nImmigrant	integer	number of immigrants	4
StopRule	integer	1=stop the search after numITR generations; 2=stop the search when there is no significant improvement	1
numITR	integer	total number of GA generations when StopRule = 1; check the accumulated improvement every numITR generations when StopRule = 2	10000
improve	real	when StopRule = 2, the value of δ in (5)	10^{-7}
SaveEvery	integer	save results every SaveEvery generations; 0=save results after completing the search	0
Nonlinear	integer	0=assume linearity; 1=incorporate nonlinear effects	0

Table 1: Input parameters: fields of the structure *Inp*; e.g., use *Inp.nSTYPE* = 2 to assign the number of stimulus types. \mathbf{I}_a is the a-by-a identity matrix.

The value of ρ is set to 0.3 and it can be changed in ‘Par_Assign.m’ through the `rho` variable. Other whitening matrices can be considered as well.

Variables	Description
<code>bestList</code>	the best design achieved
<code>bestLists</code>	the designs achieved over generations
<code>bestOVF</code>	the maximum of the objective function that we achieve
<code>OVF</code>	the values of the objective function achieved over GA generations
<code>bestidvF</code>	$[F_c^*, F_d^*, F_e^*, F_f^*]$ of the best design that we achieve
<code>idvF</code>	$[F_c^*, F_d^*, F_e^*, F_f^*]$ over GA generations
<code>timespend</code>	CPU time spent for the search (in seconds)

Table 2: Output variables: fields of the structure *Out*; e.g., `Out.bestList` contains the best design achieved.

Linear combinations of parameters

Linear combinations of the parameters of interest are specified in `Inp.CX` for model (1) and in `Inp.CZ` for model (2). These fields are, by default, set to identity matrices, allowing the study of individual stimulus effects. The number of columns for `Inp.CX` equals to the length of \mathbf{h} . For a K -second HRF (by default, $K = 32$), the length of \mathbf{h} is $Q(1 + \lfloor K/\Delta T \rfloor)$. The number of columns for `Inp.CZ` is Q , corresponding to the length of $\boldsymbol{\theta}$.

In addition to setting `Inp.CX` and `Inp.CZ` to identity matrices, researchers might also be interested in pairwise contrasts between stimulus types. [Kao et al. \(2009b\)](#) provides a systematic study of designs for convex combinations of these two interests. The example code, ‘Exp_combinedInterest.m’, is provided here for the case where equal weights are assigned to both interests.

Maximal values of the individual criteria

Values of $\max(F_e)$ and $\max(F_d)$ are assigned through `Inp.MaxFe` and `Inp.MaxFd`, respectively. These values are used to standardize F_e and F_d before combining them into an MO-criterion. Theoretical values of $\max(F_e)$ and $\max(F_d)$ are generally not available. We therefore obtain numerical approximations by performing “pre-runs” of our program. For approximating $\max(F_e)$, we use `Inp.MaxFe = 1` and `Inp.MOweight = [0 0 1 0]`; i.e., $w_e = 1$ in (3). This is equivalent to using the non-standardized F_e as the objective function. The F_e -value achieved by the design that we obtain approximates $\max(F_e)$. Similarly, we can use `Inp.MaxFd = 1` and `Inp.MOweight = [0 1 0 0]` to find the optimal design for detection and to numerically approximate $\max(F_d)$. These approximates can then be specified in `Inp.MaxFe` and `Inp.MaxFd` for further searches for multi-objective optimal designs. ‘PreRun_Fe.m’ is an example code for approximating $\max(F_e)$ and ‘PreRun_Fd.m’ is for $\max(F_d)$. The maximal values of the other two criteria, F_c and F_f , are automatically calculated in our program.

Basis for the HRF

We need to assume a basis for the HRF when using model (2). A popular choice is the canonical HRF of **SPM2** ([The Wellcome Trust Centre for Neuroimaging 2003](#)), a popular

software for fMRI. This basis is a combination of two Gamma distributions. In our program, we use this canonical HRF, scaled to have a maximum of one, as the default setting for \mathbf{h}_0 . The parameters, such as the time-to-peak and time-to-onset, used to create the canonical HRF can also be altered in ‘Par_Assign.m’. By changing `Inp.basisHRF`, other basis functions can be considered as well.

Stopping rules

We consider two stopping rules. The first stopping rule terminates the search after M_g generations. The second stopping rule is inspired by Liefvendahl and Stocki (2006). This second method considers the accumulated improvement of the design efficiency from the $\ell n + 1$ st generation to the $(\ell + 1)n$ th generation; $\ell = 0, 1, 2, \dots$. Denote the accumulated improvement by $[F^*(\xi_{(\ell+1)n}^*) - F^*(\xi_{\ell n+1}^*)]$, we stop the search at the $(\ell + 1)n$ th generation when the following condition is met (for given n and δ):

$$F^*(\xi_{(\ell+1)n}^*) - F^*(\xi_{\ell n+1}^*) \leq \delta[F^*(\xi_n^*) - F^*(\xi_1^*)]. \quad (5)$$

To use the first stopping rule, we set `Inp.StopRule = 1`, and `Inp.numITR = M_g` . By setting `Inp.StopRule = 2`, the second stopping rule is considered. In this case, we set `Inp.numITR` to n (say, 100) and `Inp.improve` to δ (e.g., 10^{-7}).

3.2. Running the code

The m-file ‘Par_Assign.m’ can directly be used to perform the search for optimal designs. With user-specified parameter values, this m-file calls the subroutine `fMRIMOD(Inp)` to start the search. Our program requires subroutines from **SPM2** (The Wellcome Trust Centre for Neuroimaging 2003) and `mttfmri` (Liu 2004b, see also Liu 2004a). From **SPM2**, we need ‘spm_Gpdf.m’ and ‘spm_hrf.m’ to calculate the canonical HRF. From `mttfmri`, we apply ‘gen_mseq.m’, ‘qadd.m’, ‘qmult.m’, ‘mseq2.m’ and ‘return_mtaps.m’ to generate m -sequence-based designs, and ‘stimpatch.m’ for plotting the final design. These m-files are freely downloadable from The Wellcome Trust Centre for Neuroimaging (2003) and Liu (2004b), respectively.

3.3. Output variables

The output variables of our program are listed in Table 2. The best design achieved by our program is contained in `Out.bestList` and its design efficiency is `Out.bestOVF`. Our program also tracks the best designs over generations (`Out.Lists`) and their design efficiencies (`Out.OVF`). The value for each individual criterion F_i^* is also provided. Time spent on the search is recorded, too.

4. An example

An illustrative example is described in this section. We consider $\text{ISI} = \text{TR} = \Delta T = 2\text{s}$, so that `Inp.ISI = 2.0`, `Inp.TR = 2.0` and `Inp.dT = 2.0`. The number of stimulus types is set to $Q = 2$ (`Inp.nSTYPE = 2`). A total of 242 events (stimuli plus the control) are included in the design sequence; i.e., `Inp.nEvents = 242`. A second-order polynomial drift and the

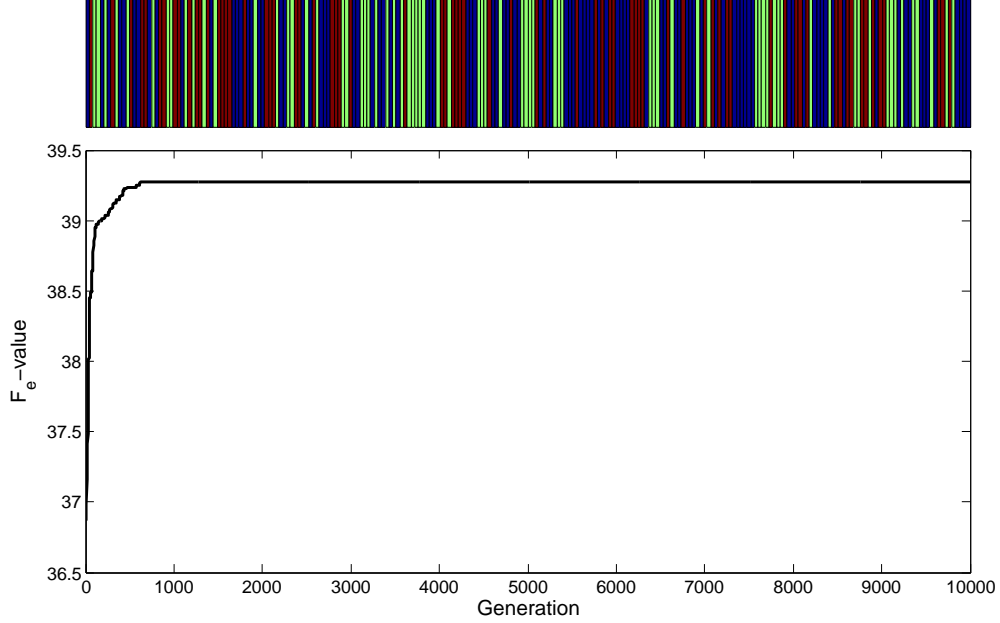


Figure 1: The design achieved by our program for estimation and F_e -values achieved over GA generations.

AR(1) noise with the correlation coefficient of 0.3 are assumed for models (1) and (2). A-optimality criterion (`Inp.Opt = 0`) is used for both statistical objectives, including estimation and detection.

The canonical HRF, scaled to have a maximum of one, is used as the basis function \mathbf{h}_0 of model (2); see also Wager and Nichols (2003); Wager *et al.* (2005). After discretization using ΔT , this basis is assigned to `Inp.basisHRF`. Following the default setting of the canonical HRF, the duration of the HRF in model (1) is $K = 32$ s, so that `Inp.durHRF = 32.0`. The number of parameters contained in each \mathbf{h}_i of model (1) is therefore $17 (= 1 + \lfloor K/\Delta T \rfloor)$, and the length of \mathbf{h} is 34. To investigate individual stimulus effects, we set `Inp.CX = eye(34)` and `Inp.CZ = eye(2)`; they are identity matrices.

For algorithmic parameters, we use the first stopping rule to terminate the search after $M_g = 10,000$ generations. Each generation consists of $G = 20$ designs. The mutation rate is $q = 1\%$ and the number of immigrants is set to $I = 4$. Therefore, `Inp.StopRule = 1`, `Inp.numITR = 10000`, `Inp.sizeGen = 20`, `Inp.qMutate = 1` and `Inp.nImmigrant = 4`.

We implement our simulations by using MATLAB (version 7.3) on a Linux cluster with 64-bit AMD Opteron, dual-processor, mix of single-core node and dual-core node; each core has 2GB RAM and the Linux operating system is 2.6.9-78.0.5.ELsmp.

We first find the (near-)optimal design for estimating the HRF by setting `Inp.M0weight = [0 0 1 0]` and `Inp.MaxFe = 1`. The resulting design is presented in Figure 1 along with the curve of the achieved efficiencies over generations. In Figure 1, each bar in the design indicates an event. Different colors represent different event types with blue indicating the

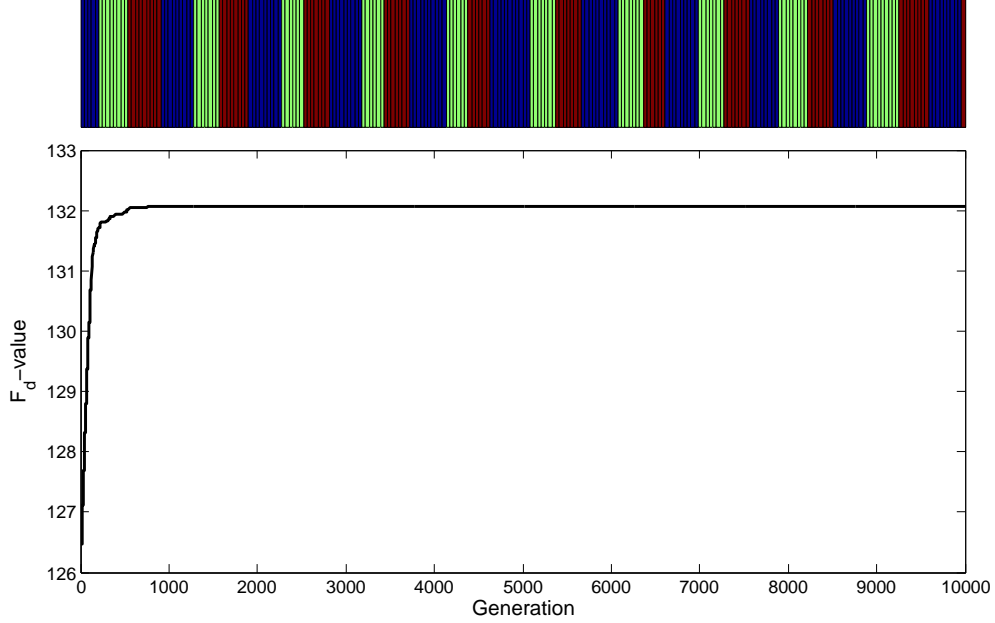


Figure 2: The design achieved by our program for detection and F_d -values achieved over GA generations.

control. This design looks rather random and its F_e -value is 39.2715. The CPU time spent for this search is 0.23 hours.

We then search for the best design for detecting activation. We set (`Inp.M0weight` = [0 1 0 0]) and `Inp.MaxFd` = 1. As presented in Figure 2, the resulting design looks like a block design. This design starts with five 0s, followed by eight stimuli of the first type and nine stimuli of the second type. The F_d -value achieved by this design is 132.0670. We spend 0.13 hours of CPU time on this search.

We can also assign equal weights to the four objectives to search for a multi-objective optimal design; i.e., `Inp.M0weight` = [1/4 1/4 1/4 1/4]. The maximal values of F_e and F_d are approximated numerically; i.e., `Inp.MaxFe` = 39.2715 and `Inp.MaxFd` = 132.0670. For F_c , we require a third-order counterbalancing property ($R = 3$), so that `Inp.cbalR` = 3. For F_f , equal frequencies for the two stimulus types are required; i.e., $P_i = 1/2$, $i = 1, 2$, and `Inp.stimFREQ` = [1/2 1/2]. Note that, when assigning `Inp.stimFREQ`, we do not take into account the number of the control event. Therefore, the actual frequency of each stimulus type in this example is less than 1/2. The number of the control event is decided by the GA search based on other requirements (design criteria). In our experience, the number of the control event is greatly influenced by the linear combinations of parameters of interest; see also, [Kao et al. \(2009b\)](#). The actual stimulus frequency of our designs agrees with the approximated optimal stimulus frequency of [Liu and Frank \(2004\)](#).

The parameter values for this last search are the same as those listed in Table 1 and those in ‘`Par_Assign.m`’. This search requires 0.42 hours of CPU time.

5. Conclusion and discussion

Optimal designs are crucial to the success of ER-fMRI experiments. Due to the nature of ER-fMRI, planning a good design is very complicated. Therefore, an efficient program that helps to find such good designs is called for. In this paper, we develop a program using MATLAB to search for multi-objective optimal experimental designs for ER-fMRI. The algorithm utilized in our program is proposed by Kao *et al.* (2009a), which is shown to outperform the previous approaches. We describe the use of our program. An example is provided for illustration. In addition to default settings, we allow the users to assign the parameter values so that our program can achieve designs best suited to the researcher's needs.

The approach considered in this article is built upon popular linear models. However, the assumption of linearity may be invalid when the stimuli are very close, and a 'saturation' in the accumulated BOLD response is observed (Wager *et al.* 2005). To take into account such a nonlinear effect, Wager and Nichols (2003) propose to use two to replace the elements of the matrix \mathbf{Z} in model (2) that are greater than two. This is also allowed in our program by setting `Inp.Nonlinear` to 1. Developing a more sophisticated method for incorporating such nonlinear effects can be useful.

Acknowledgments

Thanks are due to Professors J. Stufken and A. Mandal, N. Lazar and L. Seymour for their support and helpful discussions. The author is also thankful to two anonymous referees for their comments and suggestions, which resulted in an improvement of this work.

References

- Atkinson AC, Donev AN, Tobias RD (2007). *Optimum Experimental Designs, with SAS*. Oxford University Press, Great Britain.
- Bandettini PA, Cox RW (2000). "Event-Related fMRI Contrast When Using Constant Interstimulus Interval: Theory and Experiment." *Magnetic Resonance in Medicine*, **43**(4), 540–548.
- Buračas GT, Boynton GM (2002). "Efficient Design of Event-Related fMRI Experiments Using m-Sequences." *NeuroImage*, **16**(3), 801–813. Part 1.
- Dale AM (1999). "Optimal Experimental Design for Event-Related fMRI." *Human Brain Mapping*, **8**(2-3), 109–114.
- Friston KJ, Holmes AP, Poline JB, Grasby PJ, Williams SCR, Frackowiak RSJ, Turner R (1995). "Analysis of fMRI Time-Series Revisited." *NeuroImage*, **2**(1), 45–53.
- Holland JH (1975). *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. University of Michigan Press, Ann Arbor.

- Holland JH (1992). *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. Complex adaptive systems, 1st MIT Press edition. MIT Press, Cambridge, Massachusetts.
- Josephs O, Turner R, Friston K (1997). “Event-Related fMRI.” *Human Brain Mapping*, **5**(4), 243–248.
- Kao MH, Mandal A, Lazar N, Stufken J (2009a). “Multi-Objective Optimal Experimental Designs for Event-Related fMRI Studies.” *NeuroImage*, **44**(3), 849–856.
- Kao MH, Mandal A, Stufken J (2009b). “Optimal Design for Event-Related Functional Magnetic Resonance Imaging Considering Both Individual Stimulus Effects and Pairwise Contrasts.” *Statistics and Applications*. To appear (Special Volume in Honour of Professor Aloke Dey).
- Liefvendahl M, Stocki R (2006). “A Study on Algorithms for Optimization of Latin Hypercubes.” *Journal of Statistical Planning and Inference*, **136**(9), 3231–3247.
- Liu TT (2004a). “Efficiency, Power, and Entropy in Event-Related fMRI with Multiple Trial Types: Part II: Design of Experiments.” *NeuroImage*, **21**(1), 401–413.
- Liu TT (2004b). *mttfmri: Multiple Trial Type fMRI MATLAB Toolbox*. University of California at San Diego, Center for Functional Magnetic Resonance Imaging. URL http://cfmriweb.ucsd.edu/ttliu/mttfmri_toolbox.html.
- Liu TT, Frank LR (2004). “Efficiency, Power, and Entropy in Event-Related fMRI with Multiple Trial Types: Part I: Theory.” *NeuroImage*, **21**(1), 387–400.
- Rosen BR, Buckner RL, Dale AM (1998). “Event-Related Functional MRI: Past, Present, and Future.” *PNAS*, **95**(3), 773–780.
- The MathWorks, Inc (2006). *MATLAB – The Language of Technical Computing, Version 7.3*. The MathWorks, Inc., Natick, Massachusetts. URL <http://www.mathworks.com/products/matlab/>.
- The Wellcome Trust Centre for Neuroimaging (2003). *SPM2 – Statistical Parametric Mapping*. The Wellcome Trust Centre for Neuroimaging at the University College London, London, United Kingdom. URL <http://www.fil.ion.ucl.ac.uk/spm/software/spm2/>.
- Wager TD, Nichols TE (2003). “Optimization of Experimental Design in fMRI: A General Framework Using a Genetic Algorithm.” *NeuroImage*, **18**(2), 293–309.
- Wager TD, Vazquez A, Hernandez L, Noll DC (2005). “Accounting for Nonlinear BOLD Effects in fMRI: Parameter Estimates and a Model for Prediction in Rapid Event-Related Studies.” *NeuroImage*, **25**(1), 206–218.
- Worsley KJ, Friston KJ (1995). “Analysis of fMRI Time-Series Revisited – Again.” *NeuroImage*, **2**(3), 173–181.
- Worsley KJ, Liao CH, Aston J, Petre V, Duncan GH, Morales F, Evans AC (2002). “A General Statistical Analysis for fMRI Data.” *NeuroImage*, **15**(1), 1–15.

Affiliation:

Ming-Hung Kao
Department of Statistics
University of Georgia
Athens, GA 30602-1952, United States of America
E-mail: jasonkao@uga.edu